

# From 0 to Xamarin: Adventures in a Cross-Platform world

Fabio Cozzolino

Microsoft MVP

.NET Present and Future



Grazie a



Partner



Communities



# Chi sono



- Software Architect presso CompuGroup Medical Italia per lo sviluppo di soluzioni cloud e mobile
- Microsoft MVP dal 2010 nella categoria Integration, Microsoft Azure e Visual Studio Technologies
- Presidente di DotNetSide, community pugliese sul .NET Framework di Microsoft

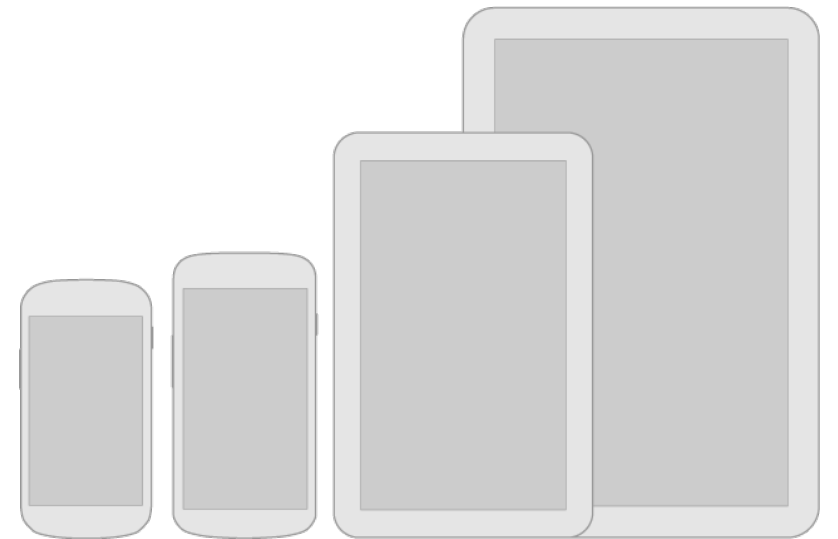
# Agenda

- Perché Cross-Platform?
- La piattaforma Xamarin
- Sviluppare per iOS e Android
- Xamarin.Forms

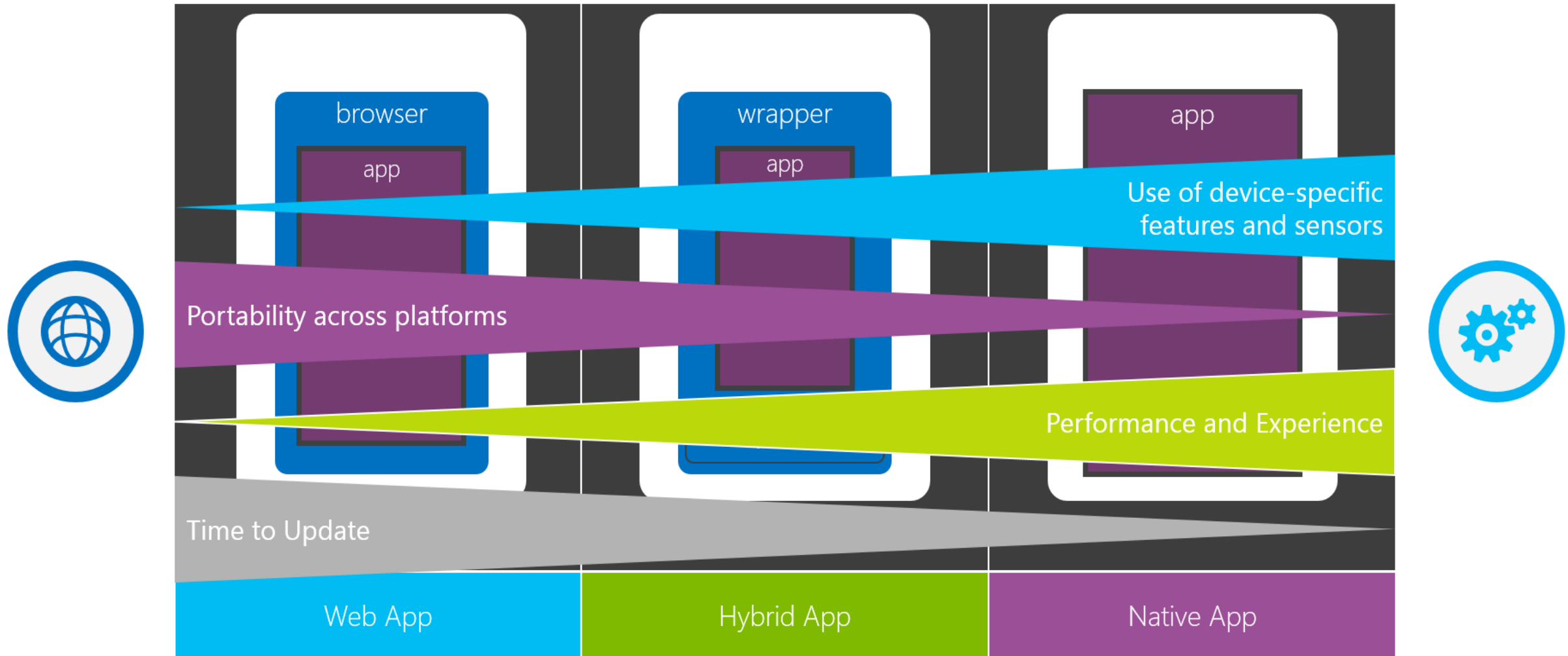
Why Cross-Platform?

# Lo sviluppo cross-platform è complicato

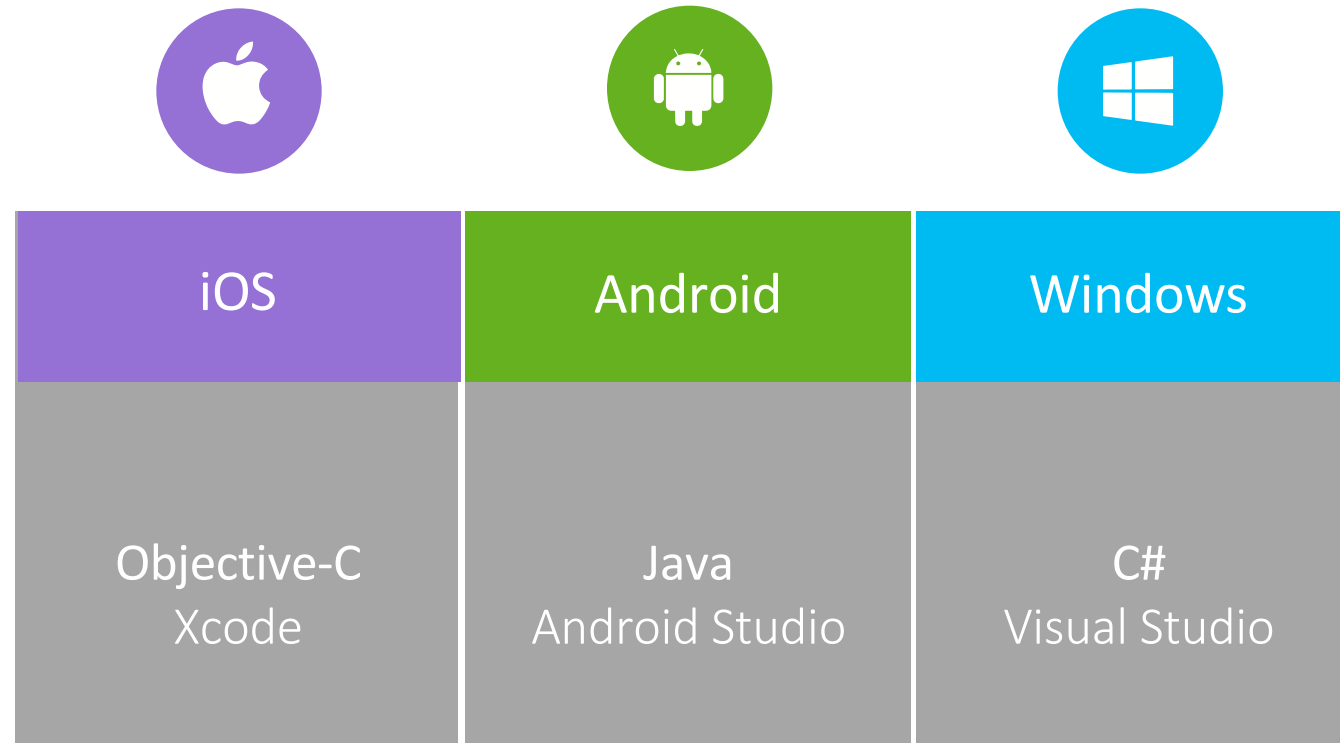
- Diversi stili di presentazione, diverse UI
- Diversi hardware, diversi input, diversi schermi
- Diversi OS e diverse versioni



# Scelta della tecnologia client



# Silo approach



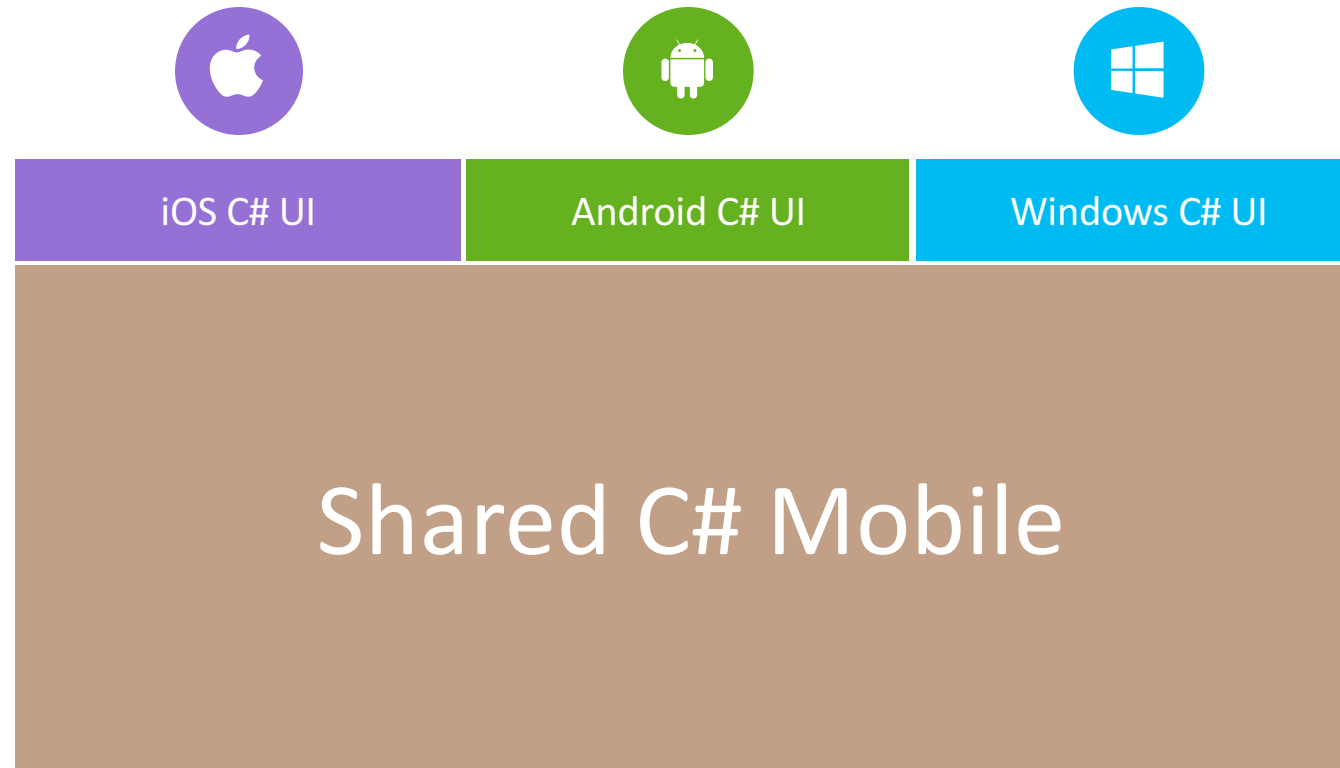
No shared code • Many languages & development environments • Multiple teams



# The Xamarin Way

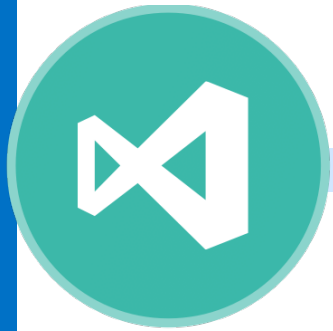
aka C# Everywhere

# Xamarin's Unique Approach



Shared C# codebase • 100% native API access • High performance

# Soluzione completa per lo sviluppo mobile



Sviluppo



Test



Build



Distribuzione



Monitoraggio

**Xamarin** test cloud > MonkeyChat > master > May 24, 2016 7:37 PM

New Test Run | [Support](#) | [Docs](#) | James ▾

Overview

ALL RESULTS

**Recorder Test**

New Test 8 ⚡

- Tapped on view with class: AppCompatActivity 6 ⚡
- Tapped on view with class: EditText
- Tapped on view with class: AppCompatActivity
- Tapped on view with class: 1 ⚡

**RECORDER TEST**  
**ScrollToEvent[AppView: Class=Xamarin.TestRec...**

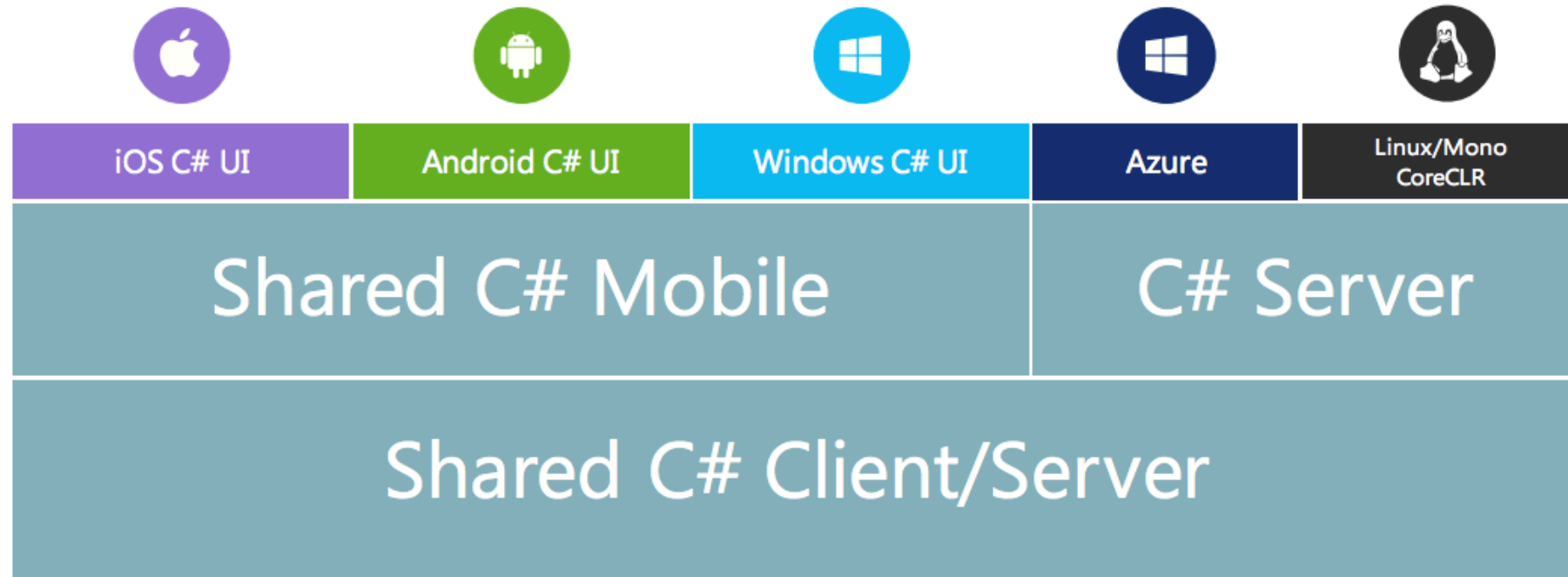
Filter devices

Device	OS
HTC One A9	Android 6.0.1
Huawei Nexus 6P	Android 6.0.1
LG Nexus 5X	Android 6.0.1
Samsung Galaxy S6	Android 5.1.1
HTC One M9	Android 5.0.2
Samsung Galaxy S6 Ed...	Android 5.0.2
Samsung Galaxy S5	Android 5.0
Samsung Galaxy Note ...	Android 4.3

# Xamarin First Touch

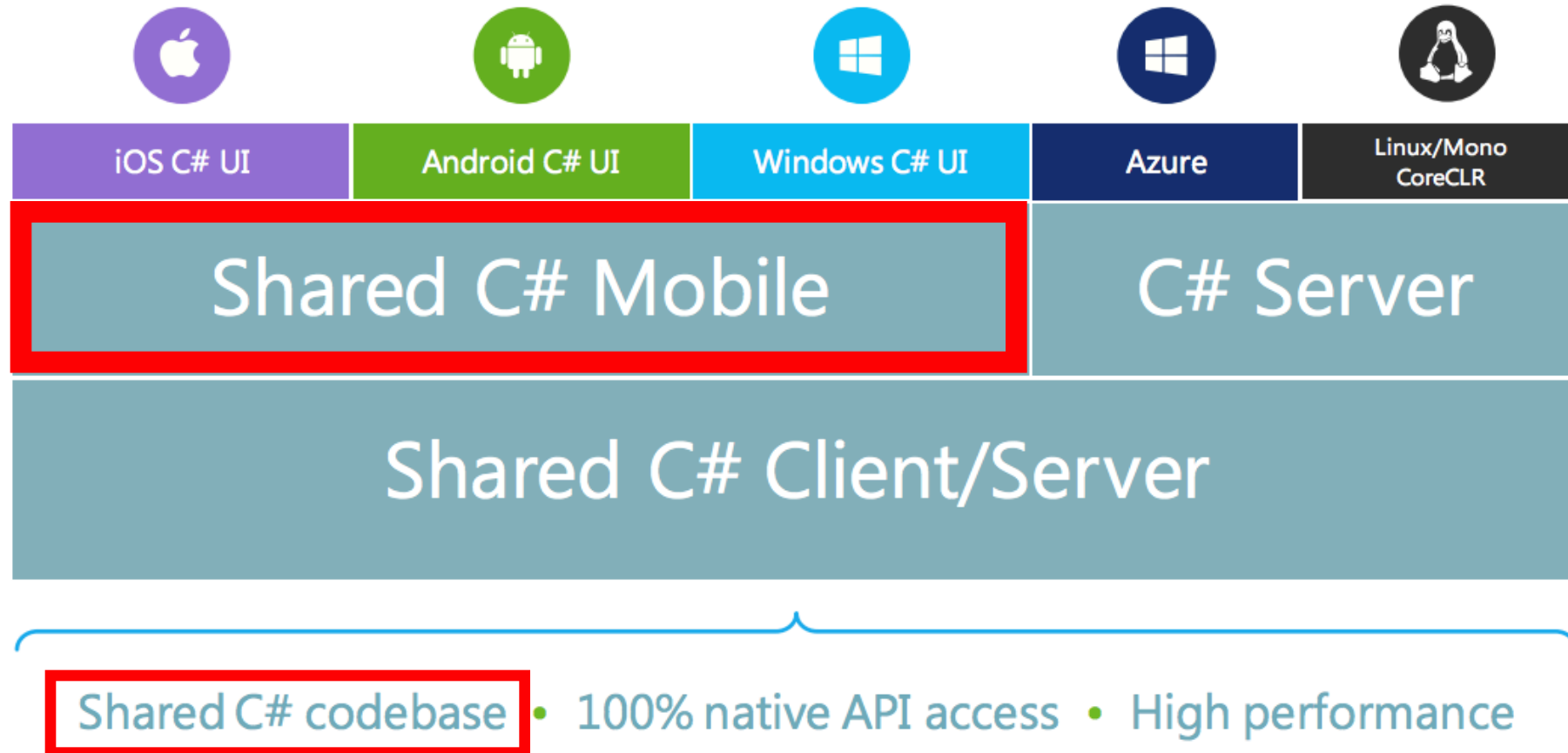
DEMO

# Sharing! Sharing! Sharing!



Shared C# codebase • 100% native API access • High performance

# Sharing! Sharing! Sharing!

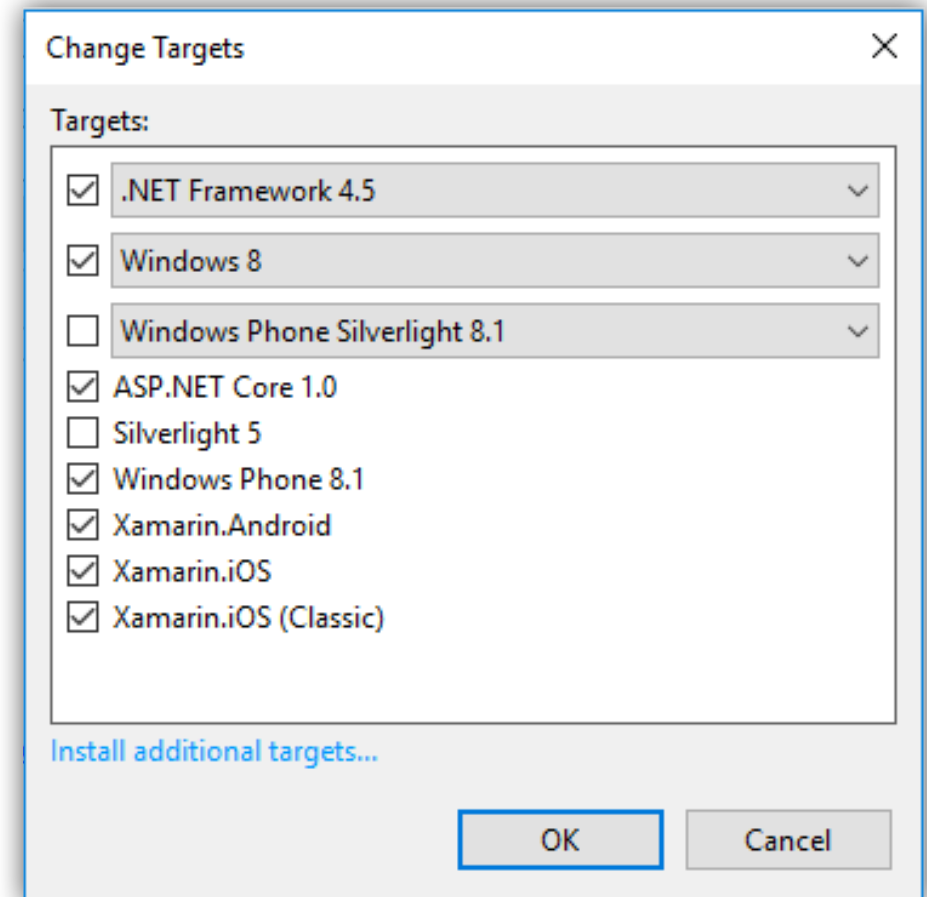


# Non son tutte rose e fiori...

- Ogni piattaforma ha le sue peculiarità
- 100% cross-platform non è possibile
- Devo gestire le specificità dell'OS
  - Ad esempio: task sulla UI
- Interazione con il device

# Condividere il codice

- Shared Project
  - Costanti di compilazione condizionale
  - Partial classes
- PCL (Portable Class Library)
  - 1 assembly per più piattaforme
  - Supporto completo in Visual Studio 2015 e in Xamarin Studio
  - codebase centralizzata

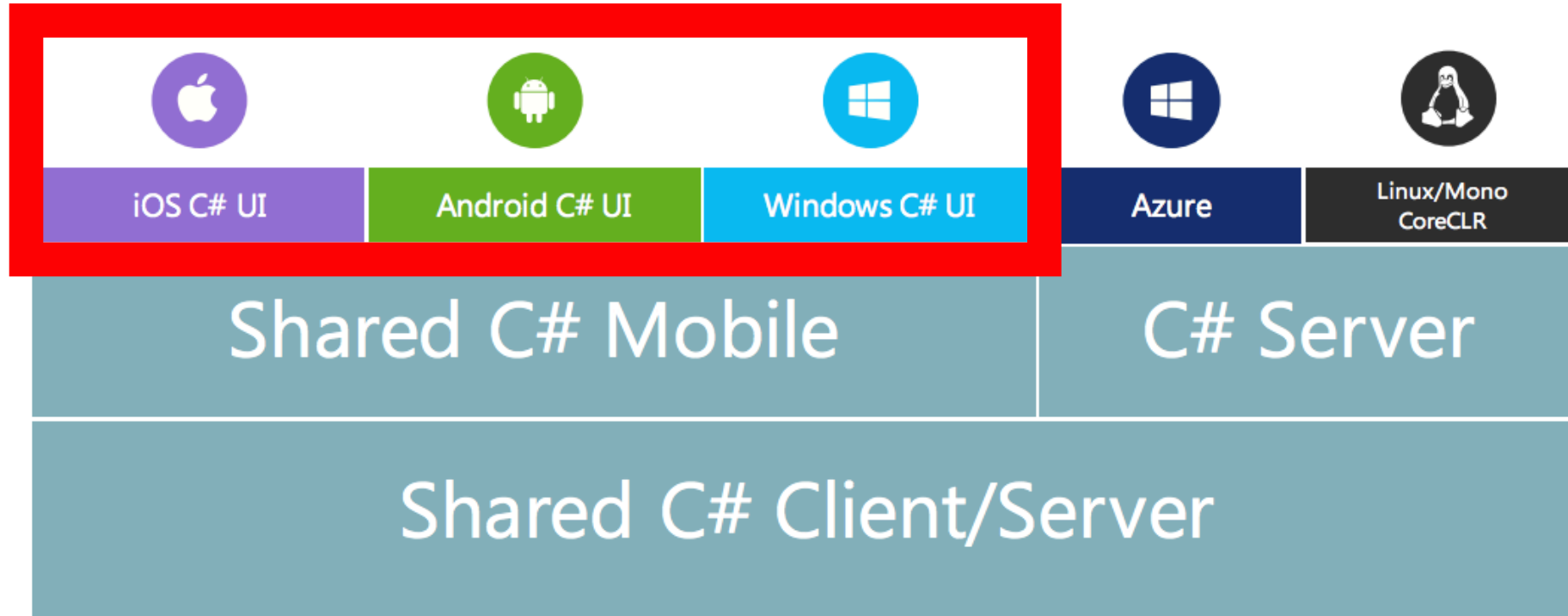




# Code sharing technique

DEMO

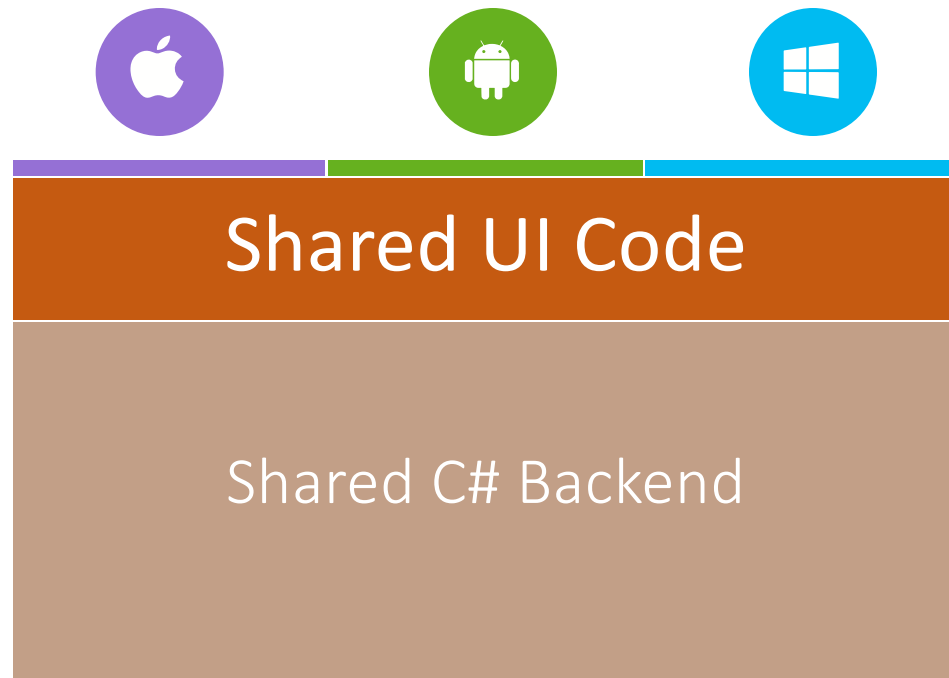
# Cross-Platform: e la UI?



Shared C# codebase • 100% native API access • High performance

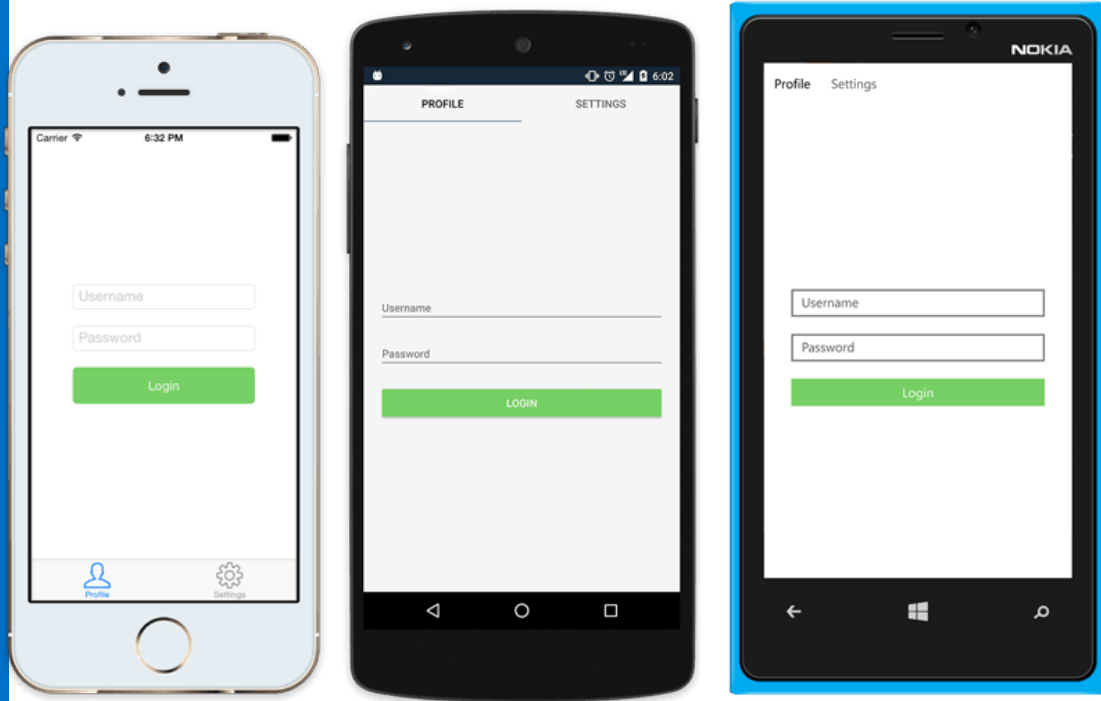
Write Once, Publish Everywhere

# What's included



- ✓ 40+ Pages, layouts, and controls  
(code behind o XAML)
- ✓ Two-way data binding
- ✓ Navigations
- ✓ Animation API
- ✓ Dependency Service
- ✓ Messaging Center

# Native UI from shared code



```
<?xml version="1.0" encoding="UTF-8"?>
<TabbedPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="MyApp.MainPage">
  <TabbedPage.Children>
    <ContentPage Title="Profile" Icon="Profile.png">
      <StackLayout Spacing="20" Padding="20"
                  VerticalOptions="Center">
        <Entry Placeholder="Username"
                Text="{Binding Username}"/>
        <Entry Placeholder="Password"
                Text="{Binding Password}"
                IsPassword="true"/>
        <Button Text="Login" TextColor="White"
                BackgroundColor="#77D065"
                Command="{Binding LoginCommand}"/>
      </StackLayout>
    </ContentPage>
    <ContentPage Title="Settings" Icon="Settings.png">
      <!-- Settings -->
    </ContentPage>
  </TabbedPage.Children>
</TabbedPage>
```

```
<ContentPage.Padding>
  <OnPlatform x:TypeArguments="Thickness">
    <OnPlatform.iOS>
      0, 20, 0, 0
    </OnPlatform.iOS>
    <OnPlatform.Android>
      0, 0, 0, 0
    </OnPlatform.Android>
    <OnPlatform.WinPhone>
      0, 0, 0, 0
    </OnPlatform.WinPhone>
  </OnPlatform>
</ContentPage.Padding>
```

Personalizzazione  
della piattaforma

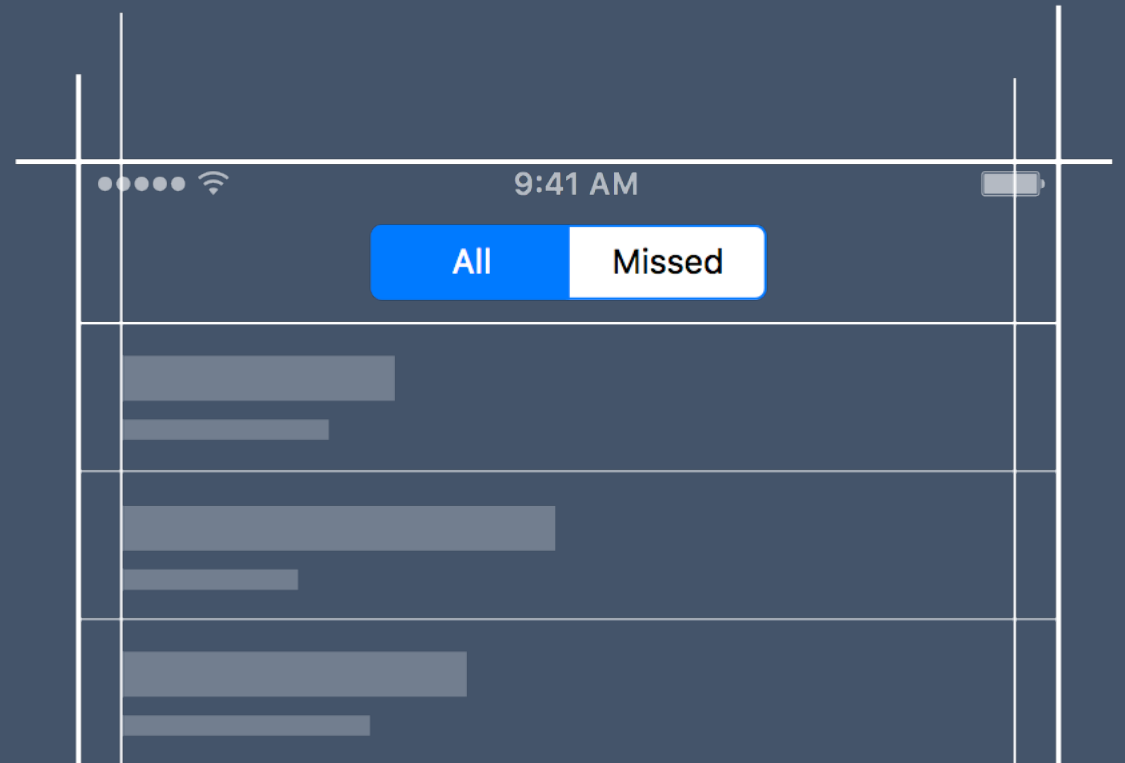
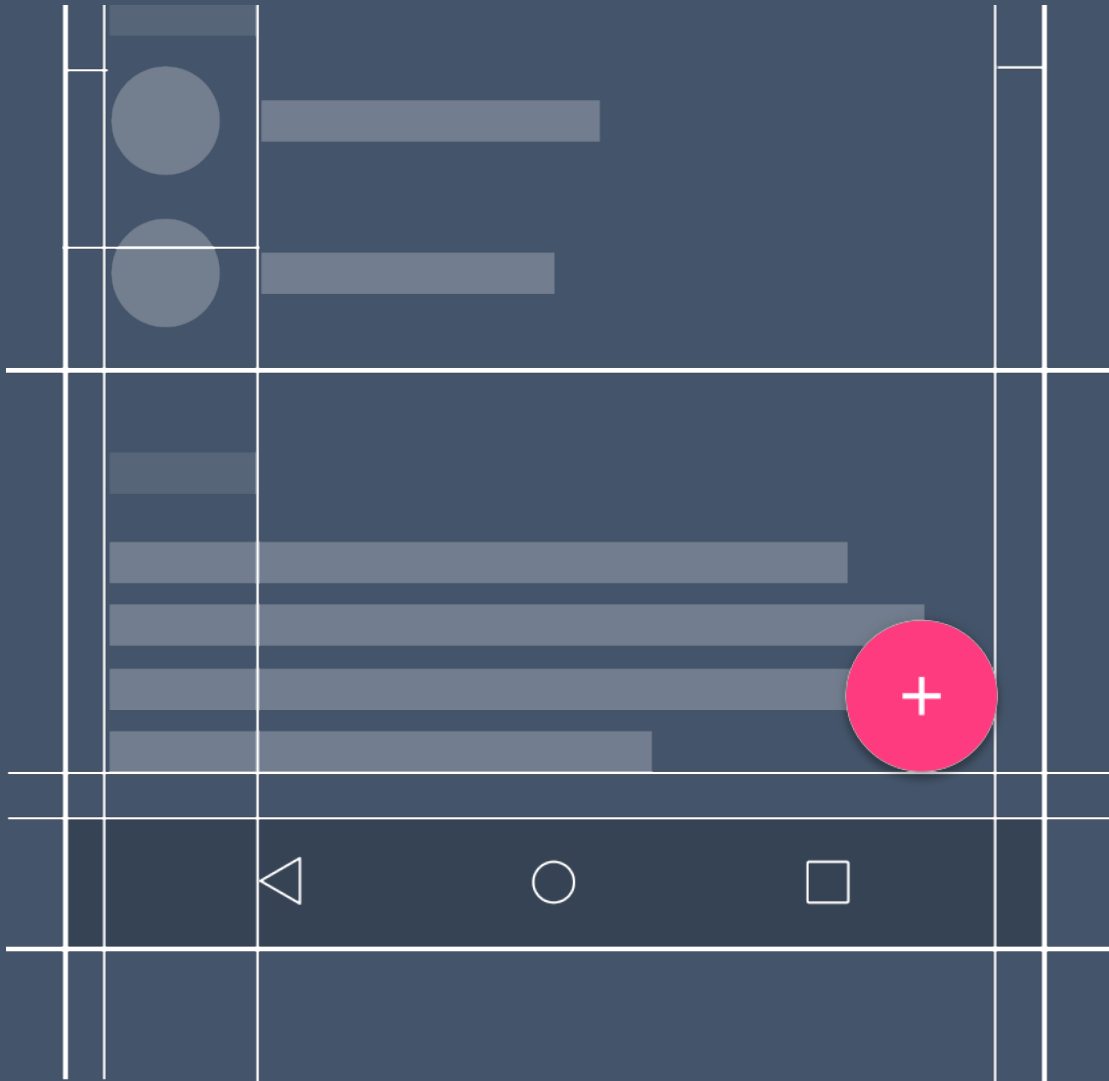
# XAML Previewer

The screenshot displays the Xamarin Studio XAML Previewer interface. The top toolbar includes a play button, a debug button, and a simulator selection dropdown currently set to 'iPhone 6s iOS 9.3'. The breadcrumb trail shows the file path: 'App.cs > ImageCircleRenderer.cs > ImageCircleRenderer.cs > CoffeesPage.xaml'. The main editor area on the left contains the following XAML code:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ContentPage
3   xmlns="http://xamarin.com/schemas/2014/forms"
4   xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
5   xmlns:local="clr-namespace:CoffeeCups;assembly=CoffeeCups"
6   xmlns:controls="clr-namespace:ImageCircle.Forms.Plugin.Abstractions;assembly=ImageCircle.Forms.Plugin.Abstractions"
7   x:Class="CoffeeCups.CoffeesPage"
8   Title="Cups Of Coffee">
9   <AbsoluteLayout HorizontalOptions="FillAndExpand" VerticalOptions="FillAndExpand">
10     <Grid AbsoluteLayout.LayoutFlags="All"
11       AbsoluteLayout.LayoutBounds="0,0,1,1"
12       RowSpacing="0">
13       <Grid.RowDefinitions>
14         <RowDefinition Height="Auto"/>
15         <RowDefinition Height="*/>
16         <RowDefinition Height="Auto"/>
17       </Grid.RowDefinitions>
18       <StackLayout Orientation="Horizontal" Padding="16" Spacing="16" BackgroundColor="Teal">
19         <Label Text="Made at Home?"
20           TextColor="White"
21           VerticalOptions="Center"
22           HorizontalOptions="Start"
23           FontSize="16">
```

The right-hand pane shows the visual preview of the application on an Android phone. The interface features a teal header bar with a toggle switch for 'Made at Home?' and an 'ADD COFFEE' button. Below the header, a list of coffee entries is displayed, each with a timestamp (e.g., '4/26/2016 10:17:53 AM'), a time (e.g., '10:17 AM'), the text 'Made At Home', and a circular image of a coffee cup. The 'Device' dropdown is set to 'Phone', and the 'Platform' is set to 'Android'.

# Native Embedding

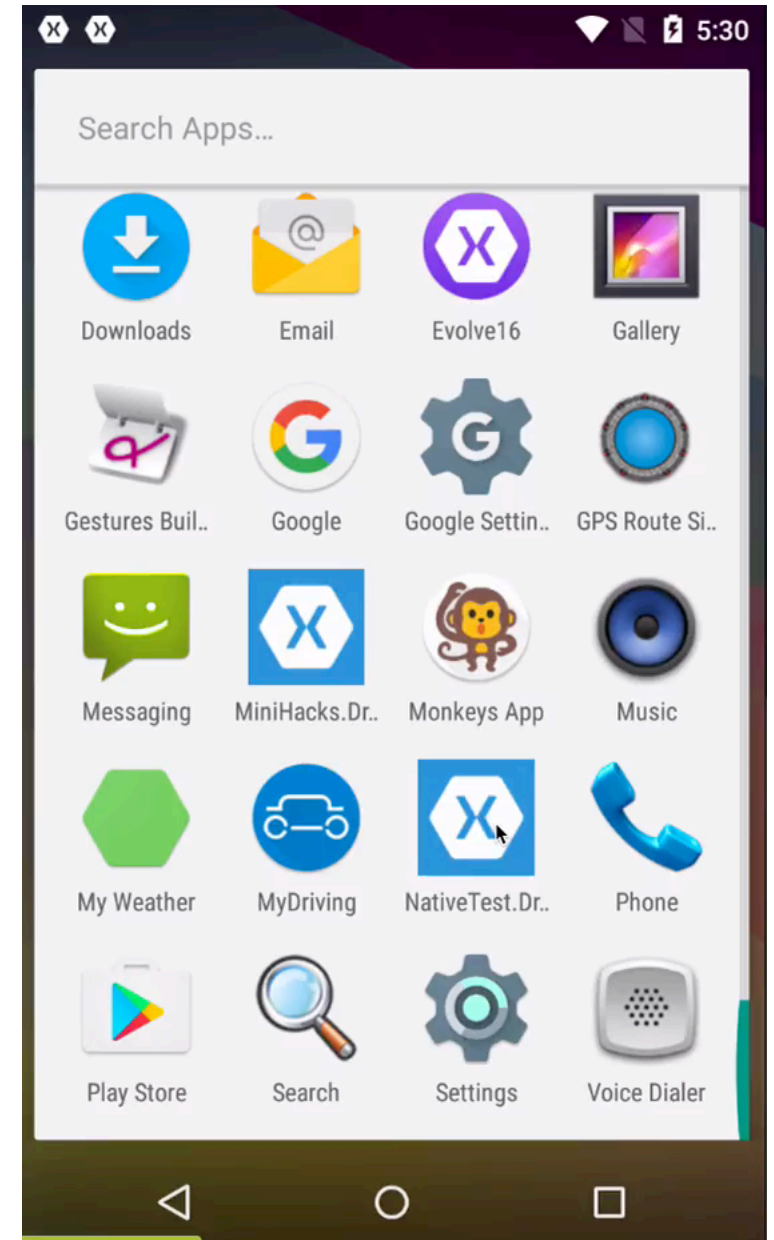




# Native Embedding

```
var stack = new StackLayout
{
    Padding = 50,
    HorizontalOptions = LayoutOptions.Center,
};

#if __ANDROID__
var fab = new CheckableFab(Forms.Context);
fab.SetImageResource(Droid.Resource.Drawable.ic_fancy_fab_icon);
fab.Click += async (sender, e) =>
{
    await MainPage.DisplayAlert("Native FAB Clicked",
                                "Whoa!!!!!!", "OK");
};
stack.Children.Add(fab);
#endif
```



# Xamarin.Forms

DEMO

# Riferimenti

- Contattatemi
  - @fabiocozzolino
  - [www.fabiocozzolino.eu](http://www.fabiocozzolino.eu)
- Xamarin
  - @xamarinhq
  - <http://www.xamarin.com>
- Sviluppatori Xamarin Italia
  - <https://www.facebook.com/groups/svilupporixamarinitalia/>

Domande?